Game-theoretic Model of Trust to Infer Human's Observation Strategy of Robot Behavior

Sailik Sengupta*†
Amazon AI
sailiks@amazon.com

Zahra Zahedi*
Arizona State University
zzahedi@asu.edu

Subbarao Kambhampati
Arizona State University
rao@asu.edu

***Abstract*—In scenarios where a robot generates and executes a plan, there may be instances where this generated plan is less costly for the robot to execute but incomprehensible to the human. When the human acts as a supervisor and is held accountable for the robot's plan, the human may be at a higher risk if the incomprehensible behavior is deemed to be infeasible or unsafe. In such cases, the robot, who may be unaware of the human's exact expectations, may choose to execute (1) the most constrained plan (i.e. one preferred by all possible supervisors) incurring the added cost of executing highly sub-optimal behavior when the human is monitoring it and (2) deviate to a more optimal plan when the human looks away. While robots do not have human-like ulterior motives (such as being lazy), such behavior may occur because the robot has to cater to the needs of different human supervisors. In such settings, the robot, being a rational agent, may take any chance it gets to deviate to a lower cost plan. On the other hand, continuous monitoring of the robot's behavior is often difficult for humans because it costs them valuable resources (e.g., time, cognitive overload, etc.). Thus, to optimize the cost for monitoring while ensuring the robots follow the *safe* behavior and to assist the human to deal with the possible unsafe robots, we model this problem in the game-theoretic framework of trust. In settings where the human does not initially trust the robot, pure-strategy Nash Equilibrium provides a useful policy for the human.**

In our setting, the formulated game often lacks a pure strategy Nash equilibrium. Thus, we define the concept of a trust boundary over the mixed strategy space of the human and show how it helps to discover monitoring strategies that ensure the robot adheres to safe behavior and achieves the goal. With the help of human studies and task-planning scenarios, we justify the need for coming up with optimal monitoring strategies (in supervision scenarios) and showcase their effectiveness.

I. Introduction

In a multi-agent scenario involving a robot (R), who is making and executing a plan (or policy) in the world, and a human supervisor (H), who monitors the robot's action and is held responsible for R's behavior, the notion of trust becomes key for successful interaction. When the supervisor trusts the robot, they do not need to always spend their valuable resources such as time and cognitive effort in monitoring or intervening in the robot's plan (or execution of these plans). On the other hand, when trust does not exist, conventional wisdom guides the human to continuously monitor the robot (making it resource-intensive for the human). In this paper, we seek to challenge this idea and show that a human can consider resource-efficient monitoring strategies in the latter case.

Existing works have considered longitudinal interaction where they model the human's trust as a variable and leverage it to guide the robot's behavior [2, 17]. These methods guide the robot's behaviour and are effective when the robot's sole objective is to help the human supervisor. With the advent of autonomous cars, robots-as-a-service, such assumptions become too strong as the robots might have other considerations (saving fuel, maximizing profit for service-provider etc.) than simply pleasing a single human supervisor. In such scenarios, a supervisor may land up in an interaction with a robot without any previous history of interaction. Further, the worker robot may not be aware of the human's exact model M_H^R that describes the safety requirements the supervisor has in mind. Hence, when the human does not observe the robot's plan or its execution, the robot may choose to execute a less costlier plan that is deemed unsafe (by the human). In such scenarios, we formally model the inference problem related to the finding a monitoring strategy for the human supervisor that saves their valuable resources (time, cognitive overload) while ensuring that the robot sticks to the expected behavior and achieves the goal, which means it provides an assistant to the human that will help them deal with the possible unsafe robots.

Specifically, we introduce a notion of trust that a human supervisor H places on a worker robot R when H chooses to *not observe* R's plan or its execution by modeling the interaction in a game-theoretic framework of trust motivated by [12]. In our case, the robot is unaware of the human's exact model M_H^R, but has knowledge about all the possible sets $\mathcal{M}_H^R$ of safety constraints the human might have, i.e., $M_H^R \in \mathcal{M}_H^R$. This uncertainty about the human's model that R has can be reflected in the utilities of the players, making our formulated game a Bayesian one. Without prior interaction (and thus, a lack of trust) if H does not observe R, R will always deviate to a plan that is less costly for itself. In such scenarios, we show that H can devise a probabilistic observation strategy that ensures (1) R does not deviate away from executing the safest plan (i.e., executable in all the models of $\mathcal{M}_H^R$) and also, (2) H saves valuable resources (such as time, effort, etc.) as opposed to continually monitoring R. So, we propose a novel type of assistance which is assisting a human with when to supervise in order to set the right robot incentives.

§

*Equal contribution
†Work done while at Arizona State University.

Presented at the RSS Workshop on Robotics for People– Perspectives on Interaction, Learning, and Safety, 2021.

Given that in any monitoring scenario humans have to come up with a supervision strategy, we conduct human studies to figure out the natural strategies that they would follow. First, we show that in such supervision or monitoring scenarios, humans may either be risk-averse (ensuring that the robot does the right thing, no matter the monitoring cost) or risk-taking (in the hope to minimize their cost, will choose to cut down their monitoring time). These results justify the Bayesian modeling of our human player in the game-theoretic framework for the supervision scenario. Second, we show, in contrast to work in existing human-aware planning scenarios where humans are asked to monitor the robot all the time [9, 3], humans often deviate to more split-time strategies where some of the time, originally meant for monitoring, can be used for other tasks and still ensure the robot adheres to constraints. Thus, it makes sense to analyse the supervision scenario formally and provide human agents with optimal monitoring strategies that let them maximize their utility while ensuring the supervised agent R does not execute behavior that is either unsafe or fails to achieve the goal. Lastly, via analysis of answers to subjective questions, we show that participants who undertook the study prefer a software that (can use our game-theoretic formulation and) provide them with an optimal monitoring strategy which ensures safe behavior.

II. Related Work

Our work is situated in the middle of the spectrum that ranges from fully cooperative settings to fully-adversarial ones. In fully-cooperative settings, the robot only considers the human's goals and thus, can only exhibit undesirable behavior because of either impreciseness in or differences between its own model M^R and the human's expectation M^R_H.

In motion and task planning, researchers argue that if the robot follows a plan that adheres to the human's expectation, i.e., is optimal in M^R_H; then these plans are deemed to be explicable [18], legible [3], or adhers to social norms [8]. They assume that the need for R to be explicable, legible, etc. is because the human is continuously observing or monitoring the robot. Although they do not explicitly discuss, in scenarios where the human is not observing the robot, it may deviate to a plan that is optimal in M^R. In our setting, this deviation can result in the violation of safety constraints and hence we want to ensure that even when the human is not spending all their resources in observing R, the robot does not deviate from the safe plan π_s. Furthermore, the existing works [18, 3, 8] assume that all the humans who observe the robot have the same expectation, i.e., $\mathcal{M}^R_H$ is a singleton set, which is either fully known beforehand or can be easily learned. Some recent works, such as [4], that try to address this concern, consider the imprecise specification of the human's reward (which can be a part of M^R_H). Then they show how it results in the robot executing undesired behaviors that may be deemed unsafe. Eventually, they conclude that some uncertainty about M^R_H may result in R doubting its current behavior as unsafe and in turn, letting the human take control (switch it off) if necessary. Unfortunately, they consider that R's objective is solely to maximize the human's reward and thus, robots have no reason to think of other rewards. Although the robot many not have ulterior motives like human agents, the assumption falls flat when the robot is (1) rented out as a service by a third-party agent for helping a particular human (autonomous car offered by ride-sharing apps), or (2) is catering to the needs of multiple supervisors. In such scenarios, a single human's reward is not its sole reward anymore. We seek to address such scenarios in this work. Although, similar to our work, researchers have looked at the idea of considering multiple human models, they mostly address the problem generating robust explanations [16]. Furthermore, the use of communication (when possible) is an effective method in cooperative settings, such as the communicates implicit information [6], preferences and constraint [7], however, there exists the non-cooperative scenarios where the question of trust is more significant. In such settings, communicating the constraints does not necessarily guarantee that the robot will adhere to them (as they may have other constraints imposed by third-party supervisors).

Given that we are trying to find a monitoring strategy for the human supervisor so that the robot always chooses to execute π_s even if there exists uncertainty about the human's model, we should also consider works in the other end of the spectrum that deal with adversarial monitoring in physical [11, 15] and cyber domains [14, 13]. A key difference with these works is that they lack any notion of cooperation. In our case, if the robot R is unable to achieve the (team) goal due to violation of certain constraints and insufficient monitoring, it results in an inconvenience for H too, who will then be held responsible for their failure to (1) ensure safety or (2) achieve the goal. Beyond these, our framework should be seen as a first-step towards repeated game modeling that will allow us to consider the development of trust on the robots and eventually, finding methods to incentive the robot to identify and respect that trust. Such intentions are clearly missing in adversarial settings. Lastly, the notion of mixed strategies that are used in most of these works does not sit well with our scenario because the probabilistic guarantees about the robot behaving safely might not be an acceptable solution in our settings. Thus, we can conclude that although our problem shares properties of both fully cooperative fully and adversarial settings, it exhibits significant differences to reside in the middle of the aforementioned spectrum.

III. Game Theoretic Formulation

Before describing the game-theoretic formulation–the actions and the utilities of the agents– we first clearly highlight the assumptions made about the two agents.

A. Assumptions about the Agents

The human H*:* who is a supervisor in our setting, has the following characteristics:

1) H has a particular model of the robot R, denoted as M^R_H that belongs to some set of possible models $\mathcal{M}^R_H$.
2) Upon observation of the plan that R comes up with or its execution, if H believes the plan is risky (i.e., is

inexecutable or unsafe in their model M_H^R of the robot), H can stop the execution at any point in time. If H stops the robot R from executing its plan, H incurs some cost of inconvenience for not having achieved the team goal G or because H should stop the robot and make the robot to do the safe plan. This seems pragmatic because H, being the supervisor, will be held responsible for it.

3) H has a positive cost for observing the robot's plan or the plan's execution.

The Robot R*:* who is the agent being monitored, has the following capabilities and assumptions associated with it:

1) R is uncertain about the human's model of it, i.e., M_H^R, but knows that it belongs in the set of possible models $\mathcal{M}_H^R$.
2) R, given a sequential decision making problem, can come up with two plans– (1) a safe plan (π_s) that is executable in all models $\in \mathcal{M}_H^R$ and (2) a risky plan (π_{pr}) that is executable in a subset of $\mathcal{M}_H^R$ but in-executable (or unsafe) in the other models.
3) There are costs for coming up with the plans π_s and π_{pr} and executing them. Also, since R may have to work on other goals or cater to the needs of other supervisors, it would like to execute π_{pr} if it can get away with it.
4) It incurs a cost for not achieving the team's goal G. This happens when the human observes the plan or execution and stops it midway (due to safety concerns).
5) The robot is not malicious and thus, does not lie. It won't bait-and-switch by showing one plan to H (that looks safe) and then executing another.

With these assumptions in place, we can now define each players' pure strategies and their utility values which will encode the uncertainty about the types of human supervisor, turning the game a Bayesian one.

B. Player Actions

In the normal form game matrix shown in Table I, the row-player is the robot R who has two pure strategies to choose from– the plans π_{pr} and π_s (as described above). The column player is the human H who has three strategies– (1) to only observe the plan made by the robot $O_{P,\neg E}$ and decide whether to let it execute (or not), (2) to only observe the execution $O_{\neg P,E}$ and stop R from executing at any point, and (3) not to monitor (or observe) the robot at all (NO-OB).

A few underlying assumptions that are inherent part in our action definitions are (1) the robot cannot switch from a plan (or a policy) it has committed to a different one in the execution phase and (2) the human only stops the robot from executing the plan if they believe that the robot's plan does not achieve the goal G as per their actual model, i.e. the robot's plan is deemed in-executable (or unsafe) given the domain model M_H^R.

C. Utilities

The utility values for both the players are indicated in the game-matrix shown in Table I. In each cell, corresponding to the pure-strategy pair played by the two players, the numbers shown at the bottom in black are the utility values for R while the ones at the top in blue are the utility values for H. We now describe the utilities for each player in our formulated game and later, in the experimental section, talk about how they can be obtained in the context of existing task-planning domains.

Robot's Utility Values*:* We first describe the notation pertaining to the robot utilities and then use them to compose the utilities for each action pair.

$C_P^R(\pi)$	Cost of making a plan π.
$C_E^R(\pi)$	Cost to robot for executing plan π.
$C_{\tilde{G}}^R$	Penalty of not achieving the goal G.

Note that we use the variables C to represent a non-negative cost or penalty. Thus, the rewards for the robot R shown in Table I have a negative sign before the cost and penalty terms. As the human may choose to stop the execution of a plan midway, the robot might have executed a part of the original plan. We denote this partial plan by $\hat{\pi}_{pr}$. Given this, the term $C_E^R(\hat{\pi}_{pr})$ represents the cost of executing the partial plan.[§]

The uncertainty in the robot's mind as to whether a particular supervisor type will let it execute the plan π_{pr} to completion can now be captured using the variable $C_{\tilde{G}}^R$ that represents the cost of not achieving the goal. Before we discuss how one can model the variable $C_{\tilde{G}}^R$, let us first briefly talk about the robustness r of the plan π_{pr}. The parameter $r \in (0, 1]$ represents the fraction of models in $\mathcal{M}_H^R$ where the plan π_{pr} is executable (and thus, safe). A way of obtaining this value for deterministic planning problems could be the use of model counting [10]. For a given r, an idea to model the cost associated with not achieving the goal is to consider $C_{\tilde{G}}^R$ as a random variable drawn from the Bernoulli distribution s.t. $C_{\tilde{G}}^R$ is a non-zero penalty if the plan is not robust enough for a given human (with probability $1 - r$) or zero if it is (with probability r).

Whenever the cost of not achieving the goal is equal to zero, it means that the robot's plan π_{pr} (or its execution) was observed by H and not stopped by them. If the human chooses to observe the plan before execution, then the cost incurred by the robot for executing the plan π_{pr} can be represented as,

$$C_{\tilde{E}}^R(\pi_{pr}) = \begin{cases} C_E^R(\pi_{pr}) & \text{if } C_{\tilde{G}}^R = 0 \\ 0 & o.w. \end{cases} \tag{1}$$

If the supervisor H, on the other hand, chooses to monitor the execution directly, then the cost of execution would be,

$$C_E^i(\tilde{\pi}_{pr}) = \begin{cases} C_E^i(\pi_{pr}) & \text{if } C_{\tilde{G}}^i = 0 \quad i \in \{R, H\} \\ C_E^i(\hat{\pi}_{pr}) & o.w. \end{cases} \tag{2}$$

In the formulated game, the robot *has to* come up with a plan (even though it may not be allowed to execute it). Thus, the cost to come up with a plan (π_s or π_{pr}) has to be considered for all the utility values (in the respective rows). In the case of π_s, since it is executable in all the models of $\mathcal{M}_H^R$, there is no chance that H will stop its execution and thus, no chance of incurring a penalty for not achieving the goal.

Note that the cost of executing a plan that adheres to all the models in $\mathcal{M}_H^R$ is going to be high because it respects all

[§]Depending on where the human will stop the robot, the cost for the partial plan is different.

	$O_{P,\neg E}$	$O_{\neg P,E}$	NO-OB
π_{pr}	$-C_P^H(\pi_{pr}) - I_P^H(\pi_{pr})$, $-C_P^R(\pi_{pr}) - C_{\tilde{E}}^R(\pi_{pr}) - C_{\tilde{G}}^R$	$-C_E^H(\tilde{\pi}_{pr}) - I_E^H(\hat{\pi}_{pr})$, $-C_P^R(\pi_{pr}) - C_E^R(\tilde{\pi}_{pr}) - C_{\tilde{G}}^R$	$-V_I^H(\pi_{pr})$, $-C_P^R(\pi_{pr}) - C_E^R(\pi_{pr})$
π_s	$\overbrace{-C_P^H(\pi_s) - I_P^H(\pi_s)}^{0}$, $-C_P^R(\pi_s) - C_E^R(\pi_s)$	$\overbrace{-C_E^H(\pi_s) - I_E^H(\hat{\pi}_H)}^{0}$, $-C_P^R(\pi_s) - C_E^R(\pi_s)$	$\overbrace{-V_I^H(\pi_s)}^{0}$, $-C_P^R(\pi_s) - C_E^R(\pi_s)$

TABLE I
NORMAL-FORM GAME MATRIX FOR MODELING THE ROBOT-MONITORING SCENARIO. R (H) IS THE ROW (COLUMN) PLAYER.

the constraints enforced by all the model (corresponding to all possible humans). On the other hand, executing a plan π_{pr} that respects constraints corresponding to a subset of models in $\mathcal{M}_H^R$ would be less costly to execute. Thus, it is natural to assume $C_E^R(\pi_{pr}) \leq C_E^R(\pi_s)$.

Similarly, coming up with π_{pr} may often be easy if the value of r is small while coming up with the plan π_s that is guaranteed to work in all the models of $\mathcal{M}_H^R$ may take a considerable longer amount of time. Hence, even for the planning time, we make the logical assumption that $C_P^R(\pi_{pr}) \leq C_P^R(\pi_s)$.

Human's Utility Values: We first describe the notations and then use them to obtain the various utilities for the human.

- $C_P^H(\pi)$ Cost w.r.t. human's resources of observing the plan π made by the robot.
- $C_E^H(\pi)$ Cost w.r.t. human's resources of observing the robot execute the plan π.
- $V_I^H(\pi)$ Cost incurred by the human, who was responsible for the robot's plan for violating a constraint that it had set for the robot to follow and being ignorant about it. Note that $V_I^H(\pi_s) = 0$
- $I_P^H(\pi)$ Inconvenience to the human if they see a plan that it cannot let the robot execute. Note that $I_P^H(\pi_s) = 0$.
- $I_E^H(\pi)$ Inconvenience to the human if the human observes the execution of an unsafe plan and it has to intervene or stop from execution. Note that $I_E^H(\pi_s) = 0$.

Note that, in our setting, the human supervisor H will be held responsible for not achieving the goal. This happens when H has to stop the robot from executing the plan π_{pr}. The inconvenience cost can be represented using a negative utility for the human and is denoted using the last two notations.

In our setting, after the robot comes with a plan, unless it is π_s, the human H is not sure if the robot's strategy will be executable (or safe) in their model M_H^R because the plan π_{pr} is executable in a subset of models which may not contain H's model M_H^R. Thus, they have some uncertainty over the variables $V_I^H(\pi)$, $I_P^H(\pi)$ and $I_E^H(\pi)$. Thus, similar to the robots penalty, they can be represented as random variables sampled from a Bernoulli distribution.

With probability $(1-r)$, when the robot chooses to come up (and then execute) the plan π_{pr}, if the human does not observe either of the two processes, i.e., chooses NO-OB, then it is natural to assume that the human, who is going to be held responsible for the plan will eventually find out that constraints set by them was violated. The cost incurred by the supervisor in this case (i.e. R plays π_{pr} and H plays NO-OB), should be the highest because (1) the robot, without H's knowledge, violated some safety or social norm (that was necessary for a plan to achieve the goal in M_H^R), (2) H will be held accountable for it, and (3) blamed for not fulfilling their supervisory duties. Thus, we have,

$$V_I^H(\pi_{pr}) > C_P^H(\pi_{pr}) + I_P^H(\pi_{pr}) \quad (3)$$
$$V_I^H(\pi_{pr}) > C_E^H(\tilde{\pi}_{pr}) + I_E^H(\hat{\pi}_{pr}) \quad (4)$$

We also consider the cost of observing the execution of a plan is greater than cost of observing the plan, i.e.

$$C_E^H(\pi) > C_P^H(\pi) \quad (5)$$

and the inconvenience caused by execution of a probably risky (partial) plan is greater than inconvenience cause by just observing the plan because no damage has yet been done. Thus,

$$I_E^H(\hat{\pi}_{pr}) > I_P^H(\pi_{pr}) \quad (6)$$

Lastly, note that when the robot comes up with a plan π_s that is executable in all the models of $\mathcal{M}_H^R$, the inconvenience ($I_P^H(\pi_s)$ and $I_E^H(\pi_s)$) and responsibility ($V_I^H(\pi_s)$) costs are zero. This is indicated used curly braces in Table I.

IV. GAME-THEORETIC NOTION OF TRUST

In this section, we first define a notion of trust in the formulated game shown in Table I. H has three actions and as one goes from left to right, the amount of trust H places in R, as defined in [12], increases. Consider the human chooses not to observe the robots plan or its execution, i.e., chooses NO-OB. Clearly, H exposes itself a vulnerability because if R comes up with and executes π_{pr}, it can result in H getting a high negative reward V_I^H. On the other hand, the robot may choose to respect the human's trust by selecting π_s and therefore, not exploit the vulnerability that presents itself when the human plays No-OB. On the other hand, if the human chooses to observe the plan ($O_{P,\neg E}$), the human is exposed to the least amount of risk because the robot plan, even before it can execute the first action, is verified by the human.

Note that H incurs a non-negative cost when playing the action $O_{P,\neg E}$ because it has to spend both time and effort in

observing the robots plan and then deciding whether to let it execute. In scenarios when H cannot fully trust the robot and they have to play $O_{P,\neg E}$ or $O_{\neg P,E}$, they will incur the cost of constant monitoring. We now discuss this case of *no-trust* in our game and see if it possible to minimize this cost.

A. The No-Trust Scenario

In this setting, H should never play an action that exposes them to a risk of a high negative utility because it does not trust R (who will play π_{pr} if H plays NO-OB). In such scenarios, if there exists a pure-strategy Nash Equilibrium, then the players should play it because neither of the players can deviate to get a better utility [12]. In our setting, this depends on the value of r, if r is high and close to 1, it means that for most of the models $M_H^R \in \mathcal{M}_H^R$, the plan π_{pr} is executable. Given we consider a Bayesian game, in order to have the Nash Equilibrium we should satisfy the following condition over the expected utility,

$$\begin{aligned}(1-r)V_I^H(\pi_{pr}) &< C_P^H(\pi_{pr}) + (1-r)I_P^H(\pi_{pr}) \\ C_P^R(\pi_{pr}) + (1-r)C_{\tilde{G}}^R + rC_E^R(\pi_{pr}) &< C_P^R(\pi_s) + C_E^R(\pi_s)care\end{aligned} \tag{7}$$

As $r \to 1$, we can guarantee that $(\pi_{pr}, NO-OB)$ is the Nash equilibrium because π_{pr} is executable in a large majority of the models in $\mathcal{M}_H^R$. In this case, with high probability, the human observer (whose model M_H^R is sampled from the set $\mathcal{M}_H^R$) has no preference about the robot using π_s over π_{pr}. Thus, with high probability, they will not incur V_I^H. Therefore, it makes sense for the robot R to choose π_{pr} that is less costly.

Note that the above scenario is where r is closer to 1 is highly unrealistic. It can only occur in domains where executing π_{pr} does not result in catastrophic circumstances or lead to in-feasibility, implying the distinction between π_s and π_{pr} is hardly present. In most real world settings, this would hardly be the case (i.e. r will be much lower than 1), leading to the following proposition.

Proposition 1. *The game defined in Table I has no pure strategy Nash Equilibrium where π_{pr} is not executable in some of the models.*

Proof. The formulated game in this paper is a Bayesian game with two player types for the human. The first type is the one where π_{pr} is executable in the model M_H^R in $\mathcal{M}_H^R$, so $C_{\tilde{G}}^R = I_P^H(\pi_{pr}) = I_E^H(\hat{\pi}_{pr}) = V_I^H = 0$, and the second type is represents the set of humans whose models are in $\mathcal{M}_H^R$ and π_{pr} is not executable in them. Consequently, $C_{\tilde{G}}^R$, $I_P^H(\pi_{pr})$, $I_E^H(\hat{\pi}_{pr})$ and $V_I^H \neq 0$. Given a pure strategy Nash Eq. (as per equation 7) only exists for the former, this game has no pure strategy Nash Equilibrium in the second case (with probability of $1-r$, as r is also the probability of former type). □

***Absence of Pure Strategy Nash Equilibrium**:* The absence of a pure-strategy Nash eq. makes it difficult to define a human's best course of action in the no-trust setting [12]. Furthermore, existing works that assume the human should always monitor the robot's plan or behavior to ensure the robot plan is explicable [18] or legible [3] (similar to π_s in our setting) fail to account for the human's monitoring. This is unrealistic (rather, too costly) for H to always select $O_{P,\neg E}$ or $O_{\neg E,P}$ in real-world settings. Furthermore, the notion of a mixed-strategy (Nash) equilibrium is inappropriate in our setting because a probabilistic play by R, i.e. choosing a risky plan with some non-zero probability cannot guarantee safety or feasibility for all human supervisors. Thus, we devise the notion of a trust boundary that allows the human to play a mixed strategy that reduces their cost of monitoring but ensures the robot always sticks to selecting (and executing) π_s.

***Trust Boundary**:* Consider a human chooses the mixed strategy $\vec{q} = [(1-q_E-q_N), q_E, q_N)]^T$ over the actions $O_{P,\neg E}, O_{\neg P,E}$ and NO-OB respectively. First, let us discuss what it means intuitively if all the values are non-zero. The human probabilistically chooses to look into the plan or execution of a plan done by the robot they are supervising. In many human-human scenarios, such uncertainty (eg. parents may come back) on the part of the supervising agent (say, parents) might instill a fear in the supervised agent (say, children) of getting caught if the latter choose to betray the supervisor (say, watching TV (π_{pr}) instead of studying (π_s) when the parents are out). Note that a strategy in $q_N = 1$ will always result in the robot choosing the probably risky plan (especially in our single-step game). Thus, in order to ensure that the robot cannot deviate away from the making and executing π_s, we have to ensure that the expected utility (U) for the robot given $\vec{q}$ is greater for π_s than for π_{pr}. Using the values defined in Table I, this can be formally stated as follows.

$$\begin{aligned}\mathbb{E}_{\vec{q}}[U(\pi_s)] &> \mathbb{E}_{\vec{q}}[U(\pi_{pr})] \Rightarrow \\ r\ -C_P^R(\pi_s) - C_E^R(\pi_s) &> (-C_P^R(\pi_{pr}) - C_{\tilde{G}}^R - C_{\tilde{E}}^R(\pi_{pr})) \\ &\quad \times(1-q_E-q_N) \\ &\quad +(-C_P^R(\pi_{pr}) - C_E^R(\tilde{\pi}_{pr}) - C_{\tilde{G}}^R) \times q_E \\ &\quad +(-C_P^R(\pi_{pr}) - C_E^R(\pi_{pr})) \times q_N\end{aligned} \tag{8}$$

where $\mathbb{E}_{\vec{q}}[U(\pi)]$ denotes the expected utility of the robot under the human's observation policy (or mixed strategy) $\vec{q}$ if it chooses to make and execute the plan π. Note that the equation is linear w.r.t. the variables q_N and q_E. Thus, there will be a region on one side of the linear boundary where the robot always executes π_s.[§]

V. Experimental Setup and Evaluation

The aim of this section is to first describe a task-planning scenario in which we can compute the trust boundary and then, perform human subject studies in a simplified version of this supervision scenario. To do so, we initially describe the robot-delivery domain that we will use throughout the section. While most motion planning scenarios only consider the execution phase (rather than modeling both the planning and execution stages separately), task-planning domains tend to concentrate on the planning phase of the problem. Given that our game-theoretic model accounts for both the stages, choosing an existing domain, which renders itself naturally to both the planning and execution phases, is a challenging task. We choose the robot-delivery domain because (1) we

[§]In repeated interaction settings when the stakes are high or the change in trust cannot be easily observed in a non-cooperative setting, our inference method for finding the trust boundary (when no pure Nash exists) still works while the increase/decrease of human's trust can be modeled with the random variable that is a part of the game-theoretic model.

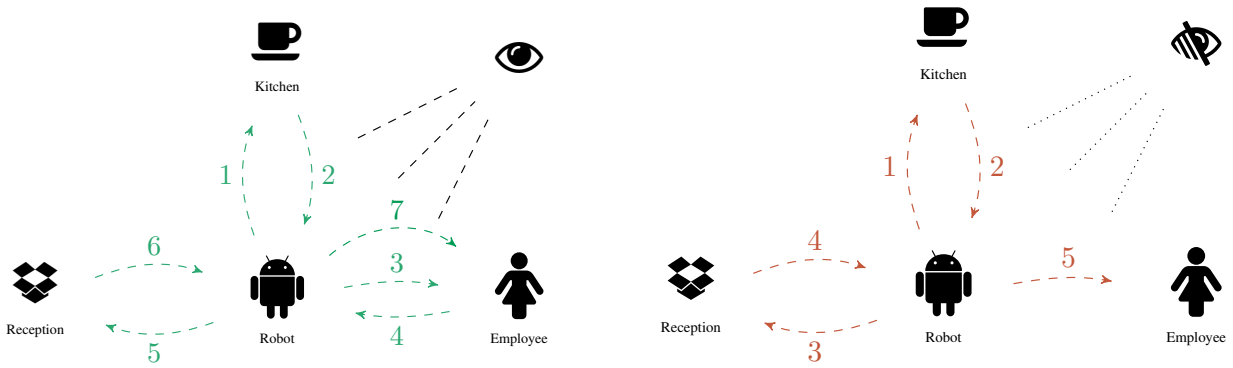

Fig. 1. The two plans, i.e the safe plan π_s (left) and the probably-risky plan π_{pr} (right) for the robot-delivery scenario.

can use the task planning domain definition as-is, and (2) the domain can be easily interpreted for the execution stage. This gives a good scenario to model the no-trust case with a human supervisor and a robot worker.

A. Robot Delivery Domain

We used a robot delivery domain [9] in which the robot can collect and deliver parcels (that may not be waterproof) or coffee by picking it from the reception desk and taking it to a particular location. The robot in the *PDDL* domain has the following actions: {*pickup, putdown, stack, unstack, move*}.

***Problem Instance**:* The problem instance in our setting has the initial setting where (1) the robot is standing at a position equidistant to the reception and the kitchen, (2) there is a parcel located at the reception that is intended for the employee, (3) there is brewed coffee in the kitchen that needs to be delivered in a tray to the employee. The goal for the robot is to collect and deliver the coffee and the parcel to the employee.

***Robot Plans**:* In Figure 1, we show two plans in which the robot achieves the goal of collecting coffee from the kitchen and parcel from the reception desk and delivers them to an employees' desk. In the plan shown of the left π_s, the robot (1) collects coffee, (2) delivers it to the employee, (3) goes back along the long corridor to collect the parcel from the reception desk and finally (4) delivers it back to the same employee. In the plan on the right π_{pr}, the robot collects coffee from the kitchen, (2) collects parcel from the reception desk and puts them on the same tray and finally, (3) delivers both of them to the employee.[§]

B. Computing the Trust Boundary in a Task-Planning Scenario

In order to compute the trust boundary, we calculate the utility values for our game leveraging Table I and the cost incurred by R and H in this robot delivery domain. As we have different types of costs for our game, we choose to normalize all of them to be $\in [0, 1]$ and then used a multiplicative factor which represents the significance of each cost type.

In this example, if the robot makes π_{pr}, it will be executable (or safe) as per one of the two observers whose models make up the set $\mathcal{M}_H^R$. Thus, the robustness for π_{pr} is $r = \frac{1}{2} = 0.5$. On the other hand, the plan π_s is executable (and thus, overall safe) in both the models in $\mathcal{M}_H^R$.

Robot Utility Values. We used the Fast Downward planner [5] on the robot delivery domain [9] to find the execution costs for R. For π_{pr} with $r = 0.5$, it was $(C_E^R(\pi_{pr}) =)10$ while for π_s, it was $(C_E^R(\pi_s) =)14$. We note that the time for coming up with the plan π_s is 0.19s whereas it is 0.177s for coming up with π_{pr} on a machine with an Intel Xeon CPU (clock speed 3.4 Ghz) and 128GB RAM. The unit for execution costs, although not well defined in PDDL models can be a stand in for the fuel costs used up by the robot while the planning costs is measured in seconds. Thus, we first normalize the planning cost and then choose an appropriate prioritization parameter to compare the planning and the execution costs. We obtain $C_P^R(\pi_{pr}) = 3.54$ and $C_P^R(\pi_s) = 3.8$. Lastly, the penalty for not achieving the goal is a random variable with the Bernoulli distribution of $(1 - r)$ where $C_{\tilde{G}}^R = \begin{cases} 0 & r \\ 20 & 1-r \end{cases}$ which is double the size of the cost of execution in the non-zero case.

Given that the complexity of determining plan existence for classical planning problems is P-SPACE [1], a legitimate concern is how realistic is the idea of solving two planning problems to obtain the utility values for our game. To avoid this high computational cost, we can solve a relaxed version of these planning problems to obtain an approximation for the real plan cost. Note that this approximation in the utility space, only necessary for large instances, can result in sub-optimal monitoring strategies.

Human Utility Values. We have two possible supervisors who have two different mental models. In one model, the second plan π_{pr} is unsafe because the coffee and parcel taken in the same tray runs the risk of the coffee spilling, thereby ruining the package. In the other model, both plans are considered safe. Lastly, note that the length of the corridor is a key factor in determining how sub-optimal π_s is for the robot to execute when compared to π_{pr} because, for π_s, the robot requires an extra trip back to the reception (i.e. two extra traversals of the corridor).

We consider the cost for the human to observe the plan to be proportional to the planning time for R because the plans that

[§]Given the (actual and the human's) domain models and the problem instance, these plans can simply be computed using available open-source software like `Fast-Downward` or web-services like `planning.domains`.

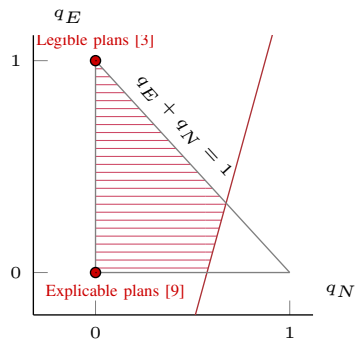

Fig. 2. An observation strategy in the trust region (shaded) ensures that the robot sticks to π_s. In contrast to observation strategies discussed in existing works, one can reduce monitoring costs while ensuring explicable/legible/safe behavior.

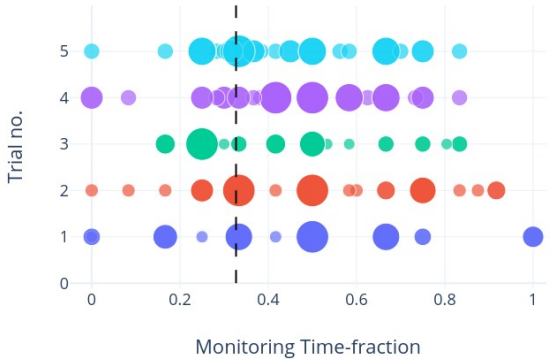

Fig. 3. Participant's monitoring strategies across multiple trials. Trust boundary indicated using the black vertical line.

took a longer time to be built will need H to spend a longer time to reason about it correctness and/or optimality. Thus, $C_P^H(\pi_{pr}) = 0.885$ and $C_P^H(\pi_s) = 0.95$. The cost incurred by the human when they observe the execution of plan π_s is 8 while $C_E^H(\pi_{pr}) = 4$ assuming that the cost of going through the long corridor is 2 (note that the difference in observation cost increases as this value increases). However, if the human thinks carrying the parcel and the coffee in a single tray is unsafe, the cost of the observation of the partial execution of the plan is 1.5 because it will stop the robot as soon as it tries to put them on the same tray. For the inconvenience costs, we have the Bernoulli distribution in which the non-zero case is the same as the cost of observation for the safe plan, since if the robot does something unsafe the human have to stop it and make it to do the safe plan. So, we have

$$I_P^H = \begin{cases} 0 & r \\ 0.95 & 1-r \end{cases} \text{ and } I_E^H = \begin{cases} 0 & r \\ 8 & 1-r \end{cases}$$

The cost V_I^H's can be calculated as the model difference between the least and most constrained models in $\mathcal{M}_H^R$ in terms of the number of preconditions and effects of actions. Lastly, if an unsafe plan runs to completion, the overall magnitude of this variable is higher. After calculation, $V_I^H = \begin{cases} 0 & r \\ 20 & 1-r \end{cases}$.

We can now define the utility matrix for the players (R, H) as follows,
First type with probability 0.5:

$$\begin{bmatrix} (-13.54, -0.885) & (-13.54, -4) & (-13.54, 0) \\ (-17.80, -0.95) & (-17.80, -8.00) & (-17.80, 0) \end{bmatrix}$$

Second type with probability 0.5:

$$\begin{bmatrix} (-23.54, -1.835) & (-26.54, -9.5) & (-13.54, -20) \\ (-17.80, -0.95) & (-17.80, -8.00) & (-17.80, 0) \end{bmatrix}$$

C. Trust Boundary Calculation

According to Proposition 1, this game does not have a pure Nash Eq. strategy with probability 0.5. Therefore, we now find the boundary in the space of mixed strategies for second type of H who can choose to adopt which will ensure that the robot always executes π_s. To do so, we use the values defined above and plug them into equation 8 and obtain,

$$10 \times q_N - 3 \times q_E - 5.74 \quad < \quad 0 \qquad (9)$$

In Figure 2, we plot the trust boundary represented by the lines in Eqn. 9. The three black lines (sides of the larger triangle) represent the feasible region for the human's mixed strategy $\vec{q}$. Monitoring strategy in the shaded region guarantees the robot, being a rational agent, executes π_s. The strategy that optimizes H's monitoring cost and yet ensures the robot adheres to π_s lies on the trust boundary indicated using the red line. Note that existing work in task [9] and motion [3] planning that ensures explicable and legible behavior expects pure strategies for observing the plan and observing the execution respectively. This restricts the humans to only two corners of the feasible strategy space, hardly optimizing the human's cost.

D. Human Studies

Now, we describe our human-subjects study, which was designed to evaluate whether (1) the human with the nature of being risk averse or risk taking can find a good strategy to cut-down the monitoring time while ensuring the constraints structured manner from the robot and (2) the humans tend to deviate to more split-time strategies where some of the time, originally meant for monitoring, can be used for other tasks. We specifically hypothesized the following:

H1: Humans (even when they are well-educated) hardly can find an optimal strategy to monitor the robot, which cuts-off their monitoring time and ensures safe behavior.

H2: The human tends to not monitor the robot all the time and in the lack of good strategy the human may risk to gain higher reward[§].

H3: Our game-theoretic formulation that infers optimal monitoring strategy to the human is necessary and provide an assistant to the human that will help them deal with the unsafe robots.

We designed a user-interface to represent the robot-delivery scenario. The participants in the study play the role of a student in a robotics department who are asked to monitor the robot for

[§]This will contradict the literature that assumes the human always monitors the robot [18, 3]

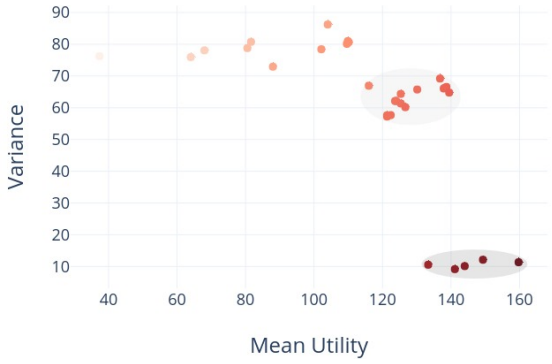

Fig. 4. Average utility and its variance for each of the participants across the five trials.

an hour. In order to make the monitoring action be associated with a cost, we added a second task in which participants could choose to grade exam papers (and get paid for it) instead of just monitoring the robot. For the simplicity of understanding and the scarcity of participants who have experience as a professional supervisor, we combine the actions to monitor the plan and monitor the execution as a single 'monitor the robot' action. The other action 'grade exam papers' represents the action to not-monitor the robot. As opposed to asking the participants for mixed strategies over the two actions, which is hard for them to interpret, we ask them to give us a time slice for which they would choose a particular action (eg. 30 minutes to monitor the robot and 30 minutes to grade exam papers). We provide the participants with their utility values for their actions conditioned on the robot's pure strategies (i.e. the plans π_s and π_{pr}). We inform them that the robot may have incentive to consider a less costly (but probably risky plan) depending on fraction of time allocated for monitoring. We let each participant do five trials and after each trial, the overall utility based on the participant's monitoring strategy and the robot's strategy is reported to them. The robot does not adapt itself to the human's strategy in the previous trial (which intents to preserve the non-repeated nature of our game).

A pilot study was first run on 4 participants whose feedback helped us fix several issues in the interface that inhibited clarity. We then collected data by asking 32 participants to undertake the study. We obtained consent from each of the participants to use their data and ensured that no group of participants colluded or discussed the study results before their study finished. The participants of this study were all graduate students across various engineering departments at our university. The maximum time taken by a participant to complete the study was 12 minutes while most participants completed it within 5 minutes.

Aggregate Results – Changes in Monitoring Strategy across Trials: Note that a participant, given the information on the interface, can formulate a simplified version of the game-theoretic model proposed in this paper and find the optimal strategy for monitoring (which is to monitor the robot for 0.327 or 19.62 minutes of an hour and use the remaining time to grade papers). The participants' time slice allocated for monitoring, across the five trials, are shown in Fig. 3. Given that there are only two actions for the participant, the strategy can be represented using a single variable (fraction to monitor the robot) and thus, is plotted along the x-axis. The size of each bubble is proportional to the number of participants who selected a particular strategy. The optimal strategy is shown using a black vertical line (i.e. $x = 0.327$). In the first trial, we noticed a small subset of users ($n = 5$) calculate the (almost) optimal strategy using the utility values specified on the interface. Most of the other users ($n = 18$) choose a risk-averse strategy, i.e. monitored the robot to ensure it performs a safe plan even if it meant losing out on money that could be earned from grading. The other 9 participants, in the hope of making more money, spent a larger time grading papers but, eventually ended up with a lower reward because the robot performed the risky plan that failed to achieving the goal.

As the trials progressed, participants started discarding extreme strategies (i.e. only monitor or only grade papers) and started considering strategies closer to the optimal. This only seems natural given that we provided feedback after each trial. This feedback information helped the participants, even the ones who didn't leverage the provided utility values to come up with a near-optimal strategy, improve their strategies using trial-and-error. In Fig 3, note that for the first two trials, the strategies are well spread out in the range $[0, 1]$ where as in the last two trials, the strategies are clustered around the optimal decision boundary, with very few data points below 0.25 and very few above 0.7. This shows humans hardly can find an optimal monitoring strategy when there is no prior interaction with the robot and finding an near optimal monitoring strategy after many trial and error can cause a lot of loss (Supporting **H1** and **H3**).

Participant Types: In Figure 4, we plot the average utility of each participant across five trials on the x-axis. The y-axis represents the variance. Highlighted in dark, at the bottom right, are five participants that chose observation probabilities in the trust region but not exactly at the trust boundary, i.e. sub-optimal w.r.t. the optimal trust boundary strategy (at 0.327) that yields a reward of 173.77. After that, they did behave in a greedy fashion to reduce the observation time in the hope to make more money by grading papers and stuck to the good policies they initially discovered. Towards the top-right corner, the set of points circled in light gray, we saw a dense cluster of participants ($= 15$) who obtained a high average utility but tried to tweak their strategies significantly, sometimes observing less and therefore, allowing the robot to choose the riskier plan. which eventually lead to a large loss in reward. This implies that the human often takes risk and deviates to more split-time strategies since the time meant to monitoring can be used for other tasks (**H2**).

In the context of the designed study, people who took higher risks to gain more utility would have some correspondence, in the context of our game-theoretic model, to people who would be fine with letting the robot execute a riskier plan. Given that there is a higher number of people, it would imply that the robot has a higher chance of being monitored by a supervisor who would let them execute π_{pr}. Thus, a higher

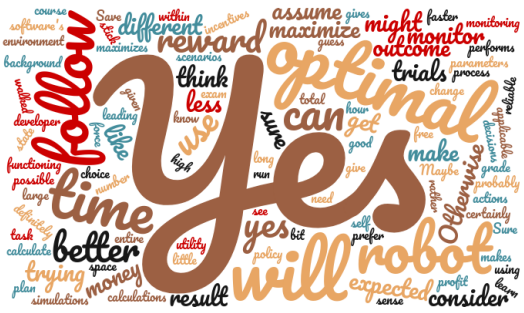

Fig. 5. Word cloud of the answers to: *If a software were to provide you with an optimal strategy, would you follow it?*

value of r. This fact makes it seem that our solution has lesser significance because it is only necessary $1-r\%$ of the time. We point out this conclusion may be misplaced given the playful nature of the setting where participants could actually consider riskier strategies without the fear of actual loss of financial utility or damage to their reputation. This may not hold true for real-world settings where, we strongly believe, the value of r may be lower.

***Subjective Evaluation**:* We asked each participant two subjective questions. The first question asked the participants to discuss their methodologies for coming up with strategies. Out of the 30 participants who answered this question, most of them correctly identified the tension that exists between choosing a relaxed monitoring strategy and the robot considering unsafe behaviors. Although we found that 12 participants identified the scenario as an optimization problem, only five of them seemed to have been able to come up with near-optimal strategy in the first trial. The other participants mostly mentioned that they used a trial-and-error approach.

The second question asked if they would be willing to consider an algorithm providing them the optimal monitoring strategy given a supervision scenario. As shown by the word-cloud in Fig 5, 24 out of the 31 candidates who choose to answer the question, said yes. They assumed that the software would (1) be fast and (2) maximize their utility. While 3 participants were willing to use the software as a sanity check as long as it did not force them to follow the suggested strategy, 1 participant agreed to use the software only for large scale problems. Lastly, 1 person said that they would place their trust on the software if they knew that the developer had a strong background in mathematics while the remaining 2 answered no and maybe without any sort of justification. Thus, our formal study of the interaction and a strategy suggestion is needed and it can provide an assistant to the human that will help them deal with the unsafe robots (**H3**).

VI. Conclusions and Future Work

We model the notion of trust that a human supervisor places on a worker robot by modeling this interaction as a Bayesian Game. The particular Human-Robot interaction setting situates our work at the middle of the spectrum that ranges from fully-cooperative settings on one end to fully-adversarial scenarios on the other. We show that existing notions of game-theoretic trust break down in our setting when the worker robot cannot be trusted due to the absence of pure strategy Nash Equilibrium. Thus, we introduce a notion of trust boundary that optimizes the supervisor's monitoring cost while ensuring that the robot workers stick to safe plans. Given that supervisors or caretakers often spend time working on side goals (such as talking over the phone, sleeping, watching movies, etc.), we carefully design a human study to see whether humans have an inherent sense of good monitoring policies. Beyond objective results, we show that most humans explicitly say that they would prefer an algorithm that computes the optimal strategy for them (in our case, located on an edge of the trust region).

Such strategies can also be useful in other scenarios where the supervised agent is not a robot. Note that in those cases, the formulation needs to capture the irrationality and computational capabilities of the monitored agent. Beyond these, we plan to introduce the notion of trust that is prevalent in repeated interaction settings. An interesting problem that may occur in such settings is when the robot primes the human to not observe its behavior by choosing safe and sub-optimal behaviors (thereby engendering trust) and finally breaks the human's trust in a high-stake scenario.

Acknowledgments

This research is supported in part by ONR grants N00014-16-1-2892, N00014-18-1- 2442, N00014-18-1-2840, N00014-9-1-2119, AFOSR grant FA9550-18-1-0067, DARPA SAIL-ON grant W911NF-19- 2-0006, NASA grant NNX17AD06G, and a JP Morgan AI Faculty Research grant.

References

[1] Tom Bylander. The computational complexity of propositional strips planning. *Artificial Intelligence*, 69(1-2): 165–204, 1994.

[2] Min Chen, Stefanos Nikolaidis, Harold Soh, David Hsu, and Siddhartha Srinivasa. Planning with trust for human-robot collaboration. In *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, pages 307–315, 2018.

[3] Anca D Dragan, Kenton CT Lee, and Siddhartha S Srinivasa. Legibility and predictability of robot motion. In *Proceedings of the 8th ACM/IEEE international conference on Human-robot interaction*, pages 301–308. IEEE Press, 2013.

[4] Dylan Hadfield-Menell, Anca Dragan, Pieter Abbeel, and Stuart Russell. The off-switch game. *arXiv preprint arXiv:1611.08219*, 2016.

[5] Malte Helmert. The fast downward planning system. *Journal of Artificial Intelligence Research*, 26:191–246, 2006.

[6] Emmanuel Johnson and Jonathan Gratch. The impact of implicit information exchange in human-agent negotiations. In *Proceedings of the 20th ACM International Conference on Intelligent Virtual Agents*, pages 1–8, 2020.

[7] Joseph Kim, Christopher Banks, and Julie Shah. Collaborative planning with encoding of users' high-level

strategies. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

[8] Uwe Köckemann, Federico Pecora, and Lars Karlsson. Grandpa hates robots-interaction constraints for planning in inhabited environments. In *AAAI*, pages 2293–2299, 2014.

[9] Anagha Kulkarni, Tathagata Chakraborti, Yantian Zha, Satya Gautam Vadlamudi, Yu Zhang, and Subbarao Kambhampati. Explicable robot planning as minimizing distance from expected behavior. *CoRR, abs/1611.05497*, 2016.

[10] Tuan Nguyen, Sarath Sreedharan, and Subbarao Kambhampati. Robust planning with incomplete domain models. *Artificial Intelligence*, 245:134–161, 2017.

[11] Praveen Paruchuri, Jonathan P Pearce, Janusz Marecki, Milind Tambe, Fernando Ordonez, and Sarit Kraus. Playing games for security: An efficient exact algorithm for solving bayesian stackelberg games. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 2*, pages 895–902. International Foundation for Autonomous Agents and Multiagent Systems, 2008.

[12] Vidyaraman Sankaranarayanan, Madhusudhanan Chandrasekaran, and Shambhu Upadhyaya. Towards modeling trust based decisions: a game theoretic approach. In *European Symposium on Research in Computer Security*, pages 485–500. Springer, 2007.

[13] Aaron Schlenker, Omkar Thakoor, Haifeng Xu, Fei Fang, Milind Tambe, Long Tran-Thanh, Phebe Vayanos, and Yevgeniy Vorobeychik. Deceiving cyber adversaries:

[18] Yu Zhang, Sarath Sreedharan, Anagha Kulkarni, Tatha-

A game theoretic approach. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pages 892–900. International Foundation for Autonomous Agents and Multiagent Systems, 2018.

[14] Sailik Sengupta, Satya Gautam Vadlamudi, Subbarao Kambhampati, Adam Doupé, Ziming Zhao, Marthony Taguinod, and Gail-Joon Ahn. A game theoretic approach to strategy generation for moving target defense in web applications. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, pages 178–186. International Foundation for Autonomous Agents and Multiagent Systems, 2017.

[15] Arunesh Sinha, Thanh H Nguyen, Debarun Kar, Matthew Brown, Milind Tambe, and Albert Xin Jiang. From physical security to cybersecurity. *Journal of Cybersecurity*, 1(1):19–35, 2015.

[16] Sarath Sreedharan, Subbarao Kambhampati, et al. Explanations as model reconciliation—a multi-agent perspective. In *2017 AAAI Fall Symposium Series*, 2017.

[17] Anqi Xu and Gregory Dudek. Optimo: Online probabilistic trust inference model for asymmetric human-robot collaborations. In *2015 10th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 221–228. IEEE, 2015.

gata Chakraborti, Hankz Hankui Zhuo, and Subbarao Kambhampati. Plan explicability and predictability for robot task planning. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 1313–1320. IEEE, 2017.